

CLAIMS

What is claimed is:

- 1 1. A method for matrix transposition, the method comprising:
2 rotating in a vector register a first row of a matrix to generate a first row of
3 elements;
4 writing simultaneously into a plurality of look up units the first row of
5 elements indexed by a first row of indices in a vector register;
6 looking up simultaneously from the plurality of look up units a second row
7 of elements indexed by a second row of indices in a vector register;
8 and
9 rotating in a vector register the second row of elements to generate a third
10 row of elements.
- 1 2. A method as in claim 1 wherein each element of the matrix comprises a
2 plurality of bit segments, each of which is written into an entry of a different
3 unit of the plurality of look up units.
- 1 3. A method as in claim 1 wherein the plurality of look up units are configured
2 into a plurality of look up tables in response to receiving an instruction for
3 looking up a row of elements.

- 1 4. A method as in claim 1 further comprising:
2 concurrently rotating in a vector register a second row of matrix to generate a
3 forth row of elements while writing the first row of elements.
- 1 5. A method as in claim 4 wherein a row that needs no rotation is written into
2 look up units before other rows are written into the look up units.
- 1 6. A method as in claim 4 further comprising:
2 concurrently computing a third row of indices using the first row of indices
3 while writing the first row of elements.
- 1 7. A method as in claim 6 further comprising:
2 concurrently loading a row of the matrix from memory into a vector register
3 while writing the first row of elements.
- 1 8. A method as in claim 6 wherein:
2 the first row of indices are a first constant;
3 the third row of indices are a second constant; and
4 the first and second constants differ by one.
- 1 9. A method as in claim 6 wherein the third row of indices is a result of a
2 rotation of the first row of indices.

- 1 10. A method as in claim 1 further comprising:
2 concurrently rotating in a vector register a fifth row of elements to generate a
3 forth row of elements while looking up the second row of elements.
- 1 11. A method as in claim 10 wherein a row of elements that needs no rotation is
2 looked up from the plurality of look up units after other rows are looked up
3 from the plurality of look up units.
- 1 12. A machine readable media containing executable computer program
2 instructions which when executed by a digital processing system cause said
3 system to perform a method for matrix transposition, the method comprising:
4 rotating in a vector register a first row of a matrix to generate a first row of
5 elements;
6 writing simultaneously into a plurality of look up units the first row of
7 elements indexed by a first row of indices in a vector register;
8 looking up simultaneously from the plurality of look up units a second row
9 of elements indexed by a second row of indices in a vector register;
10 and
11 rotating in a vector register the second row of elements to generate a third
12 row of elements.
- 1 13. A media as in claim 12 wherein each element of the matrix comprises a

1 14. A media as in claim 12 wherein the plurality of look up units are configured
2 into a plurality of look up tables in response to receiving an instruction for
3 looking up a row of elements.

1 15. A media as in claim 12 wherein the method further comprises:
2 concurrently rotating in a vector register a second row of matrix to generate a
3 forth row of elements while writing the first row of elements.

1 16. A media as in claim 15 wherein a row that needs no rotation is written into
2 look up units before other rows are written into the look up units.

1 17. A media as in claim 15 wherein the method further comprises:
2 concurrently computing a third row of indices using the first row of indices
3 while writing the first row of elements.

1 18. A media as in claim 17 wherein the method further comprises:
2 concurrently loading a row of the matrix from memory into a vector register
3 while writing the first row of elements.

1 19. A media as in claim 17 wherein:

1 20. A media as in claim 17 wherein the third row of indices is a result of a
2 rotation of the first row of indices.

1 21. A media as in claim 12 wherein the method further comprises:
2 concurrently rotating in a vector register a fifth row of elements to generate a
3 forth row of elements while looking up the second row of elements.

1 22. A media as in claim 21 wherein a row of elements that needs no rotation is
2 looked up from the plurality of look up units after other rows are looked up
3 from the plurality of look up units.

23. A processing system for matrix transposition, the system comprising:

- means for rotating in a vector register a first row of a matrix to generate a first row of elements;
- means for writing simultaneously into a plurality of look up units the first row of elements indexed by a first row of indices in a vector register;
- means for looking up simultaneously from the plurality of look up units a second row of elements indexed by a second row of indices in a vector register; and

2 a vector register file comprising a plurality of vector registers;
3 a vector processing unit coupled to the vector register file, the vector
4 processing unit comprising a vector look up unit, the vector look up
5 unit comprising a plurality of look up units adapted to look up a
6 vector of data items simultaneously, the vector processing unit:
7 rotating in a vector register in the vector register file a first row of a matrix to
8 generate a first row of elements;
9 writing simultaneously into the plurality of look up units the first row of
10 elements indexed by a first row of indices in a vector register in the
11 register file;
12 looking up simultaneously from the plurality of look up units a second row
13 of elements indexed by a second row of indices in a vector register in
14 the register file; and
15 rotating in a vector register in the vector register file the second row of
16 elements to generate a third row of elements.

1 35. A processing system as in claim 34 wherein each element of the matrix
2 comprises a plurality of bit segments, each of which is written into an entry
3 of a different unit of the plurality of look up units.

1 36. A processing system as in claim 34 wherein the plurality of look up units are
2 configured into a plurality of look up tables in response to receiving an
3 instruction for looking up a row of elements.

- 1 37. A processing system as in claim 34 wherein the vector processing unit
2 concurrently rotates in a vector register a second row of matrix to generate a
3 forth row of elements while writing the first row of elements.
- 1 38. A processing system as in claim 37 wherein a row that needs no rotation is
2 written into look up units before other rows are written into the look up units.
- 1 39. A processing system as in claim 37 wherein the vector processing unit
2 concurrently computes a third row of indices using the first row of indices
3 while writing the first row of elements.
- 1 40. A processing system as in claim 39 wherein the vector processing unit
2 concurrently loads a row of the matrix from memory into a vector register
3 while writing the first row of elements.
- 1 41. A processing system as in claim 39 wherein:
2 the first row of indices are a first constant;
3 the third row of indices are a second constant; and
4 the first and second constants differ by one.
- 1 42. A processing system as in claim 39 wherein the third row of indices is a
2 result of a rotation of the first row of indices.

1 43. A processing system as in claim 34 wherein the vector processing unit
2 concurrently rotates in a vector register a fifth row of elements to generate a
3 forth row of elements while looking up the second row of elements.

1 44. A processing system as in claim 43 wherein a row of elements that needs no
2 rotation is looked up from the plurality of look up units after other rows are
3 looked up from the plurality of look up units.